

University of Groningen

The role of the license plate lottery policy in the adoption of Electric Vehicles

Zhuge, Chengxiang; Wei, Binru; Shao, Chunfu; Shan, Yuli; Dong, Chunjiao

Published in:
Energy Policy

DOI:
[10.1016/j.enpol.2020.111328](https://doi.org/10.1016/j.enpol.2020.111328)

IMPORTANT NOTE: You are advised to consult the publisher's version (publisher's PDF) if you wish to cite from it. Please check the document version below.

Document Version
Publisher's PDF, also known as Version of record

Publication date:
2020

[Link to publication in University of Groningen/UMCG research database](#)

Citation for published version (APA):

Zhuge, C., Wei, B., Shao, C., Shan, Y., & Dong, C. (2020). The role of the license plate lottery policy in the adoption of Electric Vehicles: A case study of Beijing. *Energy Policy*, 139, [111328].
<https://doi.org/10.1016/j.enpol.2020.111328>

Copyright

Other than for strictly personal use, it is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license (like Creative Commons).

The publication may also be distributed here under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license. More information can be found on the University of Groningen website: <https://www.rug.nl/library/open-access/self-archiving-pure/taverne-amendment>.

Take-down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Downloaded from the University of Groningen/UMCG research database (Pure): <http://www.rug.nl/research/portal>. For technical reasons the number of authors shown on this cover page is limited to 10 maximum.

Reduced Order Modeling of Linear Consensus Networks using Weight Assignments

Xiaodong Cheng, Lanlin Yu, and Jacquélien M.A. Scherpen

Abstract—This paper studies a model reduction method for linear consensus networks consisting of diffusively coupled single-integrators. For a given graph clustering of an original complex network, we construct a simplified network consisting of fewer nodes, where the edge weights are to be determined. An optimal weight assignment procedure is proposed to select suitable edge weights of the reduced network, aiming for the minimum \mathcal{H}_2 approximation error between the original network and the reduced-order network model. The effectiveness of the proposed method is illustrated by means of an example.

I. INTRODUCTION

In many applications, e.g., distributed coordination, control of power grids and chemical reaction networks, consensus networks are often considered [1]–[3]. These networks with many nodes may be modeled as high-dimensional models that are too difficult to analyze and control. Therefore, it is desired to reduce a complex network model to a model of lower complexity without a significant loss of accuracy. For networks, the interconnection structures are crucial for the analysis of stability and synchronization, which are necessary for related applications e.g., distributed controller design and sensor allocation. The network systems thus have to be reduced in a manner that coupling configurations are preserved in reduced-order models. In this paper, we aim for a smaller-sized dynamical network that approximates the input-output behavior of the original network.

Classical model reduction methods, e.g., balanced truncation and moment matching [4], generally do not preserve network structures. In [5], a generalized balanced truncation is applied, allowing for a network interpretation of reduced-order model. However, the relation between the original and simplified networks is no longer clear. The mainstream approaches for simplifying dynamical networks are based on graph clustering, see e.g., [6]–[12]. The clustering-based approach roughly partitions a network into several nonoverlapping clusters (subgraphs) and then merges all the nodes in each cluster into a single node. Different approaches to select clusters are proposed. [7] suggests to choose the

almost equitable partition, a special graph clustering, for simplifying a dynamical network, and [8] develops a cluster selection method for networks with tree topology, where an asymptotically stable *edge system* can be employed to identify the importance of edges. The nodes connected by less important edges are then clustered. As a related approach, [10] establishes a bound on the approximation error based on the notation of *reducibility*, and clusters are selected to minimize that bound. In contrast, [11], [12] propose the pairwise dissimilarity of nodes in dynamical network, such that clusters in dynamical networks can be found using algorithms for static graphs. So far, all the existing clustering-based methods put their main focus on finding a suitable graph clustering, as reduced-order models are automatically generated via the Petrov-Galerkin framework.

In this paper, we also reduce the model of a network system based on graph clustering. Differently, we do not aim to find a suitable clustering and restrict our reduced order modeling to the Petrov-Galerkin framework. Instead, we explore a more accurate reduced-order model for a given clustering. To this end, a parameterized reduced-order model is thereby established for a given clustering, where the edge weights are free parameters to be determined. Then, we intend to answer the question: how to obtain a reduced-order network model with minimal approximation error by tuning the edge weights? To obtain desired edge weights, necessary and sufficient conditions in terms of matrix inequalities are derived, and then an iterative algorithm is proposed to obtain edge weights by solving the matrix inequalities. Note that the proposed scheme can be regarded as a post process for the clustering-based reduction algorithm in e.g., [12] to obtain a more accurate reduced-order network model. Finally, a numerical example is provided to illustrate the effectiveness of the proposed model reduction method.

The rest of this paper is organized as follows. In Section II, we recap some preliminaries in graph theory and introduce the problem setup. The parametric reduced-order model is formulated in Section III, and a weight assignment algorithm is proposed in Section IV, which aim to find a set of parameters to minimize the approximation error. In Section V, the proposed method is illustrated by a simulation example, and Section VI finally makes some concluding remarks.

Notation. The symbol \mathbb{R} denotes the set of real numbers. I_n is the identity matrix of size n and $\mathbf{1}_n$ represents a vector in \mathbb{R}^n of all ones. The cardinality of a set S is denoted by $|S|$. \mathcal{M} denotes the set of diagonal positive-definite matrices. For a given real matrix A , the columns of A^\perp form a basis of the null space of A , that is, $AA^\perp = 0$.

This work is supported by the European Research Council (ERC), Advanced Research Grant SYSDYNET, under the European Unions Horizon 2020 research and innovation programme (Grant Agreement No. 694504).

Xiaodong Cheng is with Control Systems Group, Department of Electrical Engineering, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands x.cheng@tue.nl

Lanlin Yu is with the Department of Automation, University of Science and Technology of China, Hefei 230026, China. lanlin.yu@rug.nl

Jacquélien M.A. Scherpen are with Jan C. Willems Center for Systems and Control, Engineering and Technology Institute Groningen, Faculty of Science and Engineering, University of Groningen, Nijenborgh 4, 9747 AG Groningen, the Netherlands. j.m.a.scherpen@rug.nl

II. PRELIMINARIES & PROBLEM SETTING

This section provides necessary definitions and concepts in graph theory used in this paper. We refer to [3], [13] for more details. The model of a dynamical network is then introduced and the model reduction problem is formulated.

A. Graph Theory

Consider a graph \mathcal{G} that consists of a finite and nonempty node set $\mathcal{V} := \{1, 2, \dots, n\}$ and an edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. If $(i, j) \in \mathcal{E}$, we say that the edge is directed from node i to node j . Thereby, the *incidence matrix* $\mathcal{B} \in \mathbb{R}^{n \times |\mathcal{E}|}$ of the directed graph is defined by

$$\mathcal{B}_{ij} = \begin{cases} +1 & \text{if edge } j \text{ is directed from node } i, \\ -1 & \text{if edge } j \text{ is directed to node } i, \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

In the case of undirected graphs, each edge is specified by an unordered pair of nodes. If each edge is assigned a positive value (weight), an undirected graph \mathcal{G} is *weighted*. For an undirected weighted graph, the Laplacian matrix is defined as follows.

$$L = \mathcal{B}W\mathcal{B}^T, \quad (2)$$

where \mathcal{B} is an incidence matrix obtained by assigning an arbitrary orientation to each edge of the undirected graph \mathcal{G} , and $W := \text{diag}(w_1, w_2, \dots, w_{|\mathcal{E}|})$ such that w_k indicates the weight associated to the edge k , for each $k = 1, 2, \dots, |\mathcal{E}|$.

Following [13], we recap the notion of graph clustering.

Definition 1: Let $\mathcal{G} := (\mathcal{V}, \mathcal{E})$ be a connected graph with a finite and nonempty node set \mathcal{V} . Then, a graph clustering is a partition of \mathcal{V} into r nonempty disjoint subsets $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_r\}$ covering all the elements in \mathcal{V} . Here, \mathcal{C}_i is called a cluster of \mathcal{G} .

The clustering of a graph can be characterized by a $(0, 1)$ -matrix, whose rows and columns are corresponding to the nodes and clusters, respectively. Specifically, we define the characteristic matrix $\Pi \in \mathbb{R}^{n \times r}$ as

$$\Pi_{ij} := \begin{cases} 1 & \text{if node } i \in \mathcal{C}_j, \\ 0 & \text{otherwise.} \end{cases} \quad (3)$$

For an undirected network, the matrix $\Pi^T L \Pi$ results in a reduced Laplacian matrix characterizing a smaller scale network [14]. Thus, for the model reduction of an undirected network, it is possible to apply the Petrov-Galerkin framework with the characteristic matrix of a selected graph clustering as the projection matrix, see e.g., [6], [7], [11], [12]. However, the reduced-order model of a network then only depends on the graph clustering, and we lose the freedom to construct a different reduced-order model from a given clustering that may result in a better approximation.

B. Problem Setup

Consider a dynamical network whose interconnection structure is characterized by a connected weighted undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ with $|\mathcal{V}| = n$. Following [12], [14], the

dynamics of the network system can be written using the Laplacian matrix of \mathcal{G} as

$$\Sigma : \begin{cases} M\dot{x} = -Lx + Fu, \\ y = Hx, \end{cases} \quad (4)$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$, and $y \in \mathbb{R}^q$ are the states, external inputs and outputs, respectively. Furthermore, M is diagonal and positive definite, and $L \in \mathbb{R}^{n \times n}$ is the Laplacian matrix characterizing the interconnection structure of the network. This paper aims to obtain a reduced-order model that not only approximates the input-output behavior of the original network system with a certain accuracy but also inherits a network structure with diffusively couplings. Thus, the following problem is addressed.

Problem 1: Given a network system Σ as in (4) and a clustering $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_r\}$, find a reduced-order model

$$\hat{\Sigma} : \begin{cases} \hat{M}\dot{\hat{x}} = -\hat{L}\hat{x} + \hat{F}u, \\ \hat{y} = \hat{H}\hat{x}, \end{cases} \quad (5)$$

with $\hat{x} \in \mathbb{R}^r$, $r \ll n$, such that the approximation error $\|\Sigma - \hat{\Sigma}\|_{\mathcal{H}_2}$ is small. $\hat{M}, \hat{L} \in \mathbb{R}^{r \times r}$, $\hat{F} \in \mathbb{R}^{r \times p}$, $\hat{H} \in \mathbb{R}^{q \times r}$ are matrices depending on the graph clustering. Moreover, $\hat{M} \in \mathcal{M}$, and \hat{L} is an undirected graph Laplacian. \square

In the later sections, we propose a novel approach to solve the problem. The approach is taken with two steps. The first step constructs a parametric reduced-order network model using graph clustering, where reduced Laplacian matrix is parameterized with unknown edge weights. Then, the second step aims to find an optimal edge weights such that the \mathcal{H}_2 -norm of approximation error bound is minimized.

III. MODELING OF REDUCED-ORDER DYNAMICAL NETWORKS

The modeling of the reduced-order model is also based on clustering. Consider an original network system of n agents as in (4). We can find a graph clustering using various methods, e.g., [8], [10], [12], [15]. Using the graph clustering $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_r\}$, we then model the reduced network system in the form of (5) with matrices \hat{M} , $\hat{L} = \hat{L}^T$, \hat{F} , and \hat{H} defined by

$$\begin{aligned} \hat{M} &= \Pi^T M \Pi, \hat{L} = \hat{\mathcal{B}} \hat{W} \hat{\mathcal{B}}^T, \\ \hat{F} &= \Pi^T F W_F, \text{ and } \hat{H} = W_H H \Pi, \end{aligned} \quad (6)$$

where $\hat{\mathcal{B}}$ is the incidence matrix of a reduced graph $\hat{\mathcal{G}}$ obtained by merging all the nodes in each cluster into a single node. $\hat{\mathcal{B}}$ actually indicates the interconnection structure of the clusters.

$$\hat{W} = \text{diag}(\hat{w}_1, \hat{w}_2, \dots, \hat{w}_m), \quad (7)$$

indicates the edge weights in the reduced network $\hat{\mathcal{G}}$, with $\hat{w}_i > 0$, $i = 1, 2, \dots, m$. $W_F \in \mathbb{R}^{p \times p}$ and $W_H \in \mathbb{R}^{q \times q}$ are diagonal matrices weighting the input and output channels of the reduced-order network system $\hat{\Sigma}$. Notice that \hat{W} , W_F and W_H are parameters that we can choose to minimize the approximation error $\|\Sigma - \hat{\Sigma}\|_{\mathcal{H}_2}$.

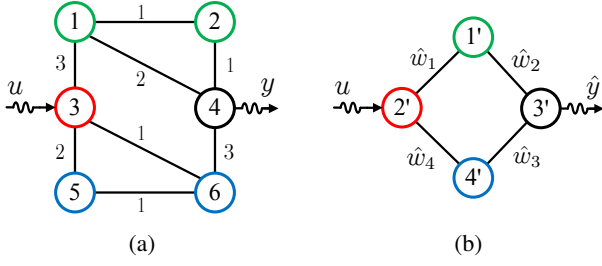


Fig. 1: (a) An undirected network consisting of 6 nodes, in which node 3 is the leader and node 4 is measured. Four clusters are indicated by different colors. (b) A clustered network consisting of 4 nodes.

The physical interpretation of the reduced-order model is further illustrated by the following example.

Example 1: Consider a network defined on an undirected graph composed of 6 nodes, see Fig. 1a. An external force u is acting on node 3, and the state of node 4 is measured as an output signal y . The incidence matrix of the graph is obtained by assigning arbitrary orientations to the edges.

$$\mathcal{B} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 \end{bmatrix}$$

We assume $M = I_6$, and the damper coefficients (edge weights) are indicated in Fig. 1a. Hereby, the network dynamics is captured by (4) with the Laplacian matrix, input and output matrices as

$$L = \begin{bmatrix} 6 & -1 & -3 & -2 & 0 & 0 \\ -1 & 2 & 0 & -1 & 0 & 0 \\ -3 & 0 & 6 & 0 & -2 & -1 \\ -2 & -1 & 0 & 6 & 0 & -3 \\ 0 & 0 & -2 & 0 & 3 & -1 \\ 0 & 0 & -1 & -3 & -1 & 5 \end{bmatrix}, F = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, H^T = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Suppose a graph clustering is given by $\mathcal{C}_1 = \{1, 2\}$, $\mathcal{C}_2 = \{3\}$, $\mathcal{C}_3 = \{4\}$, $\mathcal{C}_4 = \{5, 6\}$, which results in a reduced network, as shown in Fig. 1b, in which the incidence matrix is given by

$$\hat{\mathcal{B}} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & -1 \end{bmatrix},$$

Let $\hat{W} = \text{diag}(\hat{w}_1, \hat{w}_2, \hat{w}_3, \hat{w}_4)$ be the weights of the corresponding edges. The Laplacian matrix of the reduced graph in (6) is constructed as

$$\hat{L} = \begin{bmatrix} \hat{w}_1 + \hat{w}_2 & -\hat{w}_1 & -\hat{w}_2 & 0 \\ -\hat{w}_1 & \hat{w}_1 + \hat{w}_4 & 0 & -\hat{w}_4 \\ -\hat{w}_2 & 0 & \hat{w}_2 + \hat{w}_3 & -\hat{w}_3 \\ 0 & -\hat{w}_4 & -\hat{w}_3 & \hat{w}_3 + \hat{w}_4 \end{bmatrix},$$

with $\hat{w}_i > 0$ the parameters to be selected. $\hat{M} = \text{diag}(2, 1, 1, 2)$, $\hat{F} = W_F [0 \ 1 \ 0 \ 0]^T$, and $\hat{H} = W_H [0 \ 0 \ 1 \ 0]$. The reduced-order network system in the form of (5) can be modeled on the basis of \hat{L} , \hat{F} , and \hat{H} . \square

The error system $\Sigma_e = \Sigma - \hat{\Sigma}$ is established as follows.

$$\Sigma_e : \begin{cases} \dot{z} = - \begin{bmatrix} M^{-1}L & 0 \\ 0 & \hat{M}^{-1}\hat{L} \end{bmatrix} z + \begin{bmatrix} M^{-1}F \\ \hat{M}^{-1}\hat{F} \end{bmatrix} u, \\ y_e = [H \quad -\hat{H}] z, \end{cases} \quad (8)$$

with $z^T := [x^T \ \hat{x}^T]$. Note that Σ_e is not necessarily asymptotically stable, since the two Laplacian matrices are singular. The following lemma however shows that with constraints to W_F and W_H in (6), we ensure that $\Sigma_e \in \mathcal{H}_2$.

Lemma 1: Consider the original network system Σ in (4) and the clustered model $\hat{\Sigma}$ in (5) with parametric matrices \hat{L} , \hat{F} , and \hat{H} in (6). The approximation error $\|\Sigma - \hat{\Sigma}\|_{\mathcal{H}_2}$ is bounded if

$$\hat{F} = \beta \Pi^T F, \quad \hat{H} = \beta^{-1} H \Pi, \quad (9)$$

where β is an arbitrary nonzero scalar, and Π is the characteristic matrix of the graph clustering.

Proof: From the definition of \mathcal{H}_2 norm [4], we have

$$\|\Sigma_e\|_{\mathcal{H}_2}^2 = \int_0^\infty \text{tr} [g_e(t) g_e(t)^T] dt, \quad (10)$$

where $g_e(t)$ is the impulse response of Σ_e , i.e.,

$$g_e(t) = H e^{-M^{-1}Lt} M^{-1} F - \hat{H} e^{-\hat{M}^{-1}\hat{L}t} \hat{M}^{-1} \hat{F}. \quad (11)$$

Note that $\mathbf{1}^T M \mathbf{1} = \mathbf{1}^T \hat{M} \mathbf{1}$, $\mathbf{1}^T F = \mathbf{1}^T \hat{F}$, and $H \mathbf{1} = \hat{H} \mathbf{1}$ are sufficient conditions for asymptotic stability of the error system. Denote $\sigma_M := \mathbf{1}_n^T M \mathbf{1}_n$, which indicates the sum of the weights of the nodes in the original network. It is straightforward to verify that $\mathbf{1}_r^T \hat{M} \mathbf{1}_r^T = \sigma_M$. Then, following [12], [14], we have

$$\begin{aligned} \lim_{t \rightarrow \infty} g_e(t) &= \sigma_M^{-1} [H \quad -\hat{H}] \begin{bmatrix} \mathbf{1}_n \mathbf{1}_n^T M & 0 \\ 0 & \mathbf{1}_r \mathbf{1}_r^T \hat{M} \end{bmatrix} \begin{bmatrix} M^{-1} F \\ \hat{M}^{-1} \hat{F} \end{bmatrix} \\ &= \sigma_M^{-1} (H \mathbf{1}_n \mathbf{1}_n^T F - \hat{H} \mathbf{1}_r \mathbf{1}_r^T \hat{F}). \end{aligned}$$

If (9) holds, i.e., $W_F = \beta I_p$ and $W_H = \beta^{-1} I_q$, we obtain

$$\hat{H} \mathbf{1}_r \mathbf{1}_r^T \hat{F} = \beta H \mathbf{1}_n \mathbf{1}_n^T F \beta^{-1} = c H \mathbf{1}_n \mathbf{1}_n^T F. \quad (12)$$

Thus, $\lim_{t \rightarrow \infty} g_e(t) = 0$, which implying that $\Sigma_e \in \mathcal{H}_2$. \blacksquare

In this paper, \hat{M} , \hat{F} , and \hat{H} are not used as optimization variables, since the problem could be hard to solve due to the coupled variables, and thus it is beyond the scope of this work. By choosing the input and output matrices \hat{F} and \hat{H} as (9), we obtain the following transfer function

$$\hat{G}(s) = H \Pi (s \hat{M} + \hat{L})^{-1} \Pi^T F. \quad (13)$$

It is worth emphasizing that the input-output behavior of $\hat{\Sigma}$ is invariant to the choice of scalar β in (9). The error $\|\Sigma - \hat{\Sigma}\|_{\mathcal{H}_2}$ only depends on the choices of \hat{L} , or more precisely, the diagonal weight matrix \hat{W} in (6). Hereby, we further address the following problem in the next section.

Problem 2: Consider the original network system Σ in (4). For a given graph clustering, a reduced network model $\hat{\Sigma}$ is obtained with the transfer function (13). Find a diagonal positive definite matrix $\hat{W} \in \mathbb{R}^{m \times m}$ such that $\|\Sigma - \hat{\Sigma}\|_{\mathcal{H}_2}$ is minimized. \square

IV. WEIGHT ASSIGNMENT

In this section, we propose an iterative method for selecting appropriate weights of the edges in the reduced network, namely the parametric matrix \hat{W} in (6). Note that the error system Σ_e in (8) is semistable, whose \mathcal{H}_2 norm cannot be computed directly using its Gramian matrices. For that, we consider matrices

$$S_n = \begin{bmatrix} -I_{n-1} \\ \mathbf{1}_{n-1}^T \end{bmatrix} \in \mathbb{R}^{n \times (n-1)}, S_r = \begin{bmatrix} -I_{r-1} \\ \mathbf{1}_{r-1}^T \end{bmatrix} \in \mathbb{R}^{r \times (r-1)} \quad (14)$$

which satisfy $S_n^T \mathbf{1}_n = 0$ and $S_r^T \mathbf{1}_r = 0$, respectively, and their left pseudo inverses are denoted by

$$S_n^\dagger := (S_n^T M^{-1} S_n)^{-1} S_n^T M^{-1} \in \mathbb{R}^{(n-1) \times n}, \quad (15)$$

$$S_r^\dagger := (S_r^T \hat{M}^{-1} S_r)^{-1} S_r^T \hat{M}^{-1} \in \mathbb{R}^{(r-1) \times r}.$$

Now, we denote

$$\mathcal{U}_e = \begin{bmatrix} \sigma_M^{-1} \mathbf{1}_n & 0 & M^{-1} S_n & 0 \\ 0 & \sigma_M^{-1} \mathbf{1}_r & 0 & \hat{M}^{-1} S_r \end{bmatrix}, \mathcal{U}_e^{-1} = \begin{bmatrix} \mathbf{1}_n^T M & 0 \\ 0 & \mathbf{1}_r^T \hat{M} \\ S_n^T M & 0 \\ 0 & S_r^T \hat{M} \end{bmatrix},$$

where $\sigma_M = \mathbf{1}_n^T M \mathbf{1}_n = \mathbf{1}_r^T \hat{M} \mathbf{1}_r$. Let $z = \mathcal{U}_e \tilde{z}$ in (8), which then leads to

$$\tilde{\Sigma}_e : \begin{cases} \dot{\tilde{z}} = - \begin{bmatrix} 0 & & \\ & 0 & \\ & & L_s \end{bmatrix} \tilde{z} + [S_r^\dagger \hat{F}] u, \\ y_e = \begin{bmatrix} \sigma_M^{-1} H \mathbf{1}_n & -\sigma_M^{-1} \hat{H} \mathbf{1}_r \\ H M^{-1} S_n & -\hat{H} \hat{M}^{-1} S_r \end{bmatrix} \tilde{z}, \end{cases} \quad (16)$$

where $L_s = S_n^\dagger L M^{-1} S_n$, and $\hat{L}_s = S_r^\dagger \hat{B} \hat{W} \hat{B}^T \hat{M}^{-1} S_r$. Consider \hat{F} and \hat{H} in (9), we obtain the transfer function

$$G_e(s) = H M^{-1} S_n (sI + L_s)^{-1} S_n^\dagger F - H \Pi (\Pi^T M \Pi)^{-1} S_r (sI + \hat{L}_s)^{-1} S_r^\dagger \Pi^T F, \quad (17)$$

which corresponds to the following state- space representation

$$\tilde{\Sigma}_e : \begin{cases} \dot{\tilde{z}} = A_e \tilde{z} + B_e u, \\ y_e = C_e \tilde{z}, \end{cases} \quad (18)$$

where $\tilde{z} := \begin{bmatrix} S_n^\dagger M & 0 \\ 0 & S_r^\dagger \hat{M} \end{bmatrix} z \in \mathbb{R}^{n+r-2}$, and

$$A_e := - \begin{bmatrix} L_s & \\ & \hat{L}_s \end{bmatrix}, \quad B_e := \begin{bmatrix} S_n^\dagger F \\ \beta S_r^\dagger \Pi^T F \end{bmatrix},$$

$$C_e := [H M^{-1} S_n \quad -\beta^{-1} H \Pi (\Pi^T M \Pi)^{-1} S_r].$$

Note that the model in (18) is obtained by the modes corresponding to the poles at the origin in (16). A_e in (18) is Hurwitz, namely, the system in (18) is asymptotically stable.

Remark 1: Since the nonzero scalar β is canceled out in the transfer function of the error system in (17), the choices of β will not affect the approximation error. We choose S_r in the form of (14) to facilitate the follow-up optimization process, since S_r is independent from \hat{W} . If the reduced order r and the graph clustering are given, \hat{W} is the only unknown parameter to be determined. \square

Lemma 2: Consider the asymptotically stable error system (18). $\|G_e(s)\|_2 < \gamma$ holds if and only if there exist matrices $Q \in \mathbb{R}^{(n+r-2) \times (n+r-2)}$, $Q = Q^T > 0$, $R \in \mathbb{R}^{q \times q}$, $R = R^T > 0$, such that the following matrix inequalities are satisfied:

$$\begin{bmatrix} Q A_e + A_e^T Q & Q B_e \\ B_e^T Q & -I_p \end{bmatrix} < 0, \quad (19)$$

$$\begin{bmatrix} Q & C_e^T \\ C_e & R \end{bmatrix} > 0, \quad (20)$$

$$\text{tr}(R) < \gamma, \quad (21)$$

where A_e, B_e, C_e are defined in (18).

Proof: A similar proof can be found in e.g., [16]. \blacksquare

The following theorem provides necessary and sufficient conditions for the construction of a solution to Problem 2.

Theorem 1: Given $\gamma > 0$, r ($2 \leq r < n$), and the network system Σ in (4). There exists a reduced-order network system $\tilde{\Sigma}$ in (5) such that the asymptotically stable error system (18) satisfies $\|G_e(s)\|_2 < \gamma$ if and only if there exist matrices $Q \in \mathbb{R}^{(n+r-2) \times (n+r-2)}$, $Q = Q^T > 0$, $R \in \mathbb{R}^{q \times q}$, $R = R^T > 0$, $K \in \mathbb{R}^{(2(n+r-2)+p) \times (r-1)}$, $\hat{W} \in \mathcal{M}$, such that the inequalities (20) and (21) hold and the following inequality is satisfied.

$$T^T Z T + T^T K \Sigma + (T^T K \Sigma)^T < 0, \quad (22)$$

where

$$T = \begin{bmatrix} [0_{(n-1) \times (r-1)} & -L_s & 0_{(n-1) \times (r-1)} & S_n^\dagger F] \\ & I_{(n-1)+2(r-1)+p} & & \end{bmatrix},$$

$$\Sigma = \begin{bmatrix} -I_{r-1} & 0_{(r-1) \times (n-1)} & -\hat{L}_s & \beta S_r^\dagger \Pi^T F \end{bmatrix},$$

$$Z = \begin{bmatrix} 0 & Q & 0 \\ Q & 0 & 0 \\ 0 & 0 & -I_p \end{bmatrix}.$$

Proof: The inequality (19) can be rewritten as

$$(V^\perp)^T Z V^\perp < 0, \text{ with } V^\perp = \begin{bmatrix} A_e & B_e \\ I_{n+r-2} & 0 \\ 0 & I_p \end{bmatrix}, \quad (23)$$

the columns of V^\perp form a basis of the null space of matrix $V = [-I_{n+r-2} \quad A_e \quad B_e]$.

According to Finsler's Lemma [17], the inequality (23) is equivalent to

$$Z + \mathcal{K} \mathcal{V} + (\mathcal{K} \mathcal{V})^T < 0, \quad (24)$$

where $\mathcal{K} \in \mathbb{R}^{(2(n+r-2)+p) \times (n+r-2)}$. Partitioning \mathcal{K} and \mathcal{V} into blocks, yields

$$\mathcal{K} = [\bar{K} \quad K], \quad \mathcal{V} = \begin{bmatrix} \bar{V} \\ V \end{bmatrix}, \quad (25)$$

where $\bar{V} = [-I_{n-1} \quad 0 \quad -L_s \quad 0 \quad S_n^\dagger F]$, and $V = [0 \quad -I_{r-1} \quad 0 \quad -\hat{L}_s \quad S_r^\dagger \Pi^T F]$. Combining (25) and (24) leads to $Z + \bar{K} \bar{V} + K V + (\bar{K} \bar{V} + K V)^T < 0$, which, according to Finsler's Lemma, is equivalent to

$$(\bar{V}^\perp)^T Z \bar{V}^\perp + (\bar{V}^\perp)^T K V \bar{V}^\perp + ((\bar{V}^\perp)^T K V \bar{V}^\perp)^T < 0. \quad (26)$$

Choosing $\bar{V}^\perp \triangleq T$, one obtains that $\Sigma = VT$. Thus, the inequality (26) can be rewritten as (22), which implies that the inequalities (19) and (22) are equivalent. \blacksquare

Based on Theorem 1, Problem 2 can be transformed as the following optimization problem

$$\begin{aligned} \min \quad & \gamma \\ \text{s.t.} \quad & (20) - (22) \\ & \hat{W} \in \mathcal{M}, Q > 0, R > 0. \end{aligned} \quad (27)$$

Remark 2: We use (22) instead of (19) in Lemma 2, since Q in (22) is not coupled with \hat{L}_s . It is beneficial for solving the optimization problem (27). The inequality (22) is not linear with respect to the parameters Q , \hat{W} and K . However, if the parameter K is fixed, the inequality (22) becomes linear. Then, the optimization problem (27) is convex and can be solved efficiently. \square

The following iterative algorithm is provided to solve the optimization problem (27).

Algorithm 1 H_2 Model Reduction for Network Systems

Input: $S_n, S_n^\dagger, S_r, S_r^\dagger, \Pi, \beta, m > 0, 1 \leq r < n, \varepsilon > 0, \hat{B} \in \mathbb{R}^{r \times m}, L, F, H$.

Output: $\hat{W} \in \mathcal{M}, \hat{L}, \gamma^*$.

- 1: **Initialization**
 - 2: Let $L_s = S_n^\dagger L M^{-1} S_n$, $\hat{M} = \Pi^T M \Pi$, and $C_e = [H M^{-1} S_n \quad -\beta^{-1} H \Pi \hat{M}^{-1} S_r]$. Set $i = 1$ and $\gamma^{(0)}$ to be a sufficiently large number. Choose some initial $\hat{W}^{(0)} \in \mathcal{M}$ for matrix \hat{W} .
 - 3: **repeat**
 - 4: Fix $\hat{W} = \hat{W}^{(i-1)}$ and $\gamma = \gamma^{(i-1)}$, solve the inequalities (20)-(22) for Q, R and K . Denote $K^{(i)}$ and $R^{(i)}$ as the obtained K and R .
 - 5: Fix $K = K^{(i)}$, solve the following optimization problem for Q, \hat{W}, R in (27). Denote $\hat{W}^{(i)}$ as the obtained \hat{W} . Set $\gamma^{(i)} = \text{tr}(R^{(i)})$ and $i = i + 1$.
 - 6: **until** $|\gamma^{(i)} - \gamma^{(i-1)}| \leq \varepsilon$ or $i \geq i^*$ with i^* being the maximum number iterations, output $\hat{W}^{(i)}$,
- $$\gamma^* = \gamma^{(i)}, \quad \hat{L} = \hat{B} \hat{W}^{(i)} \hat{B}^T.$$
-

Remark 3: In Algorithm 1, for the fixed K , the optimization problem is convex and can be solved effectively. With the increasing iteration numbers, γ is guaranteed to be reduced, that is, it is expected to reduce the approximation error bound as the iteration number increases.

Remark 4: In the following section, we compare our method with the clustering-based projection method in [7], in which the considered model is a special case of (4) with $M = I$. The resulting reduced-order model in [7], $\hat{x} = -(\Pi^T \Pi)^{-1} \Pi^T L \Pi + (\Pi^T \Pi)^{-1} \Pi^T F u$, can be rewritten in the form of (5) with $\hat{M} = \Pi^T \Pi$.

V. ILLUSTRATIVE EXAMPLE

In this section, an example is provided to illustrate the effectiveness of the proposed model reduction method for the network systems. We make the comparisons with the projection-based method in [7]. The optimization problem is solved by the LMI solver YALMIP [18].

Following [7], we consider a network system consisting of 10 nodes which are linearly diffusively coupled. External inputs are injected into the nodes 6 and 7. The network topology is illustrated in Fig. 2. The Laplacian and the input matrices are given by

$$L = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & -5 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & -3 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 6 & -1 & -2 & -3 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 6 & -5 & 0 & 0 & 0 & 0 & 0 \\ 0 & -3 & -2 & -5 & 25 & -2 & -6 & -7 & 0 & 0 \\ -5 & -2 & -3 & 0 & -2 & 25 & -6 & -7 & 0 & 0 \\ 0 & 0 & 0 & 0 & -6 & -6 & 15 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & -7 & -7 & -1 & 15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix}, \quad F = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix},$$

We choose the same graph clustering as in [7], where an almost equitable partition is adopted, i.e.,

$$\begin{aligned} \mathcal{C}_1 &= \{1, 2, 3, 4\}, & \mathcal{C}_2 &= \{5, 6\}, & \mathcal{C}_3 &= \{7\}, \\ \mathcal{C}_4 &= \{8\}, & \mathcal{C}_5 &= \{9, 10\}, \end{aligned}$$

A reduced network is then obtained by aggregating all the nodes in each cluster. The interconnection structure of the reduced network is shown in Fig. 1b, whose incidence matrix is formulated as

$$\hat{B} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 1 \\ 0 & 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{bmatrix}.$$

Different from [7], we model the reduced-order network model with parametric weights $\hat{w}_1, \hat{w}_2, \hat{w}_3, \hat{w}_4, \hat{w}_5$. In the rest of this section, we choose different output matrices for the original network and compare the reduction results of the proposed methods in the current paper and [7].

Case 1. Choosing the output matrix as $H = [0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ -1]$, which is different with the output given in [7]. Note that in this case, the error bound in [7] does not hold any more. For comparison, we reduce the original network using the projection-based model reduction method in [7] and the proposed method in this paper, respectively, while we choose the same graph clustering for both methods. In our method, we let $\beta = 1$. By using Algorithm 1, a sub-optimal \hat{W} can be obtained as $\hat{W} = \text{diag}(11.7976, 9.7164, 13.156, 0.03258, 2.0232)$, resulting in a reduced-order network system in the form of (5) with matrices $\hat{M} = \text{diag}(4, 2, 1, 1, 2)$, and

$$\begin{aligned} \hat{L} &= \begin{bmatrix} 11.7976 & -11.7976 & 0 & 0 & 0 \\ -11.7976 & 21.5465 & -9.7164 & -0.0325 & 0 \\ 0 & -9.7164 & 24.8956 & -13.1560 & -2.0232 \\ 0 & -0.0325 & -13.156 & 13.1885 & 0 \\ 0 & 0 & -2.0232 & 0 & 2.0232 \end{bmatrix}, \\ \hat{F} &= [0 \ 0 \ 0 \ 0 \ 0]^T, \quad \hat{H} = [0 \ 1 \ 0 \ 0 \ -1]. \end{aligned}$$

Moreover, the relative \mathcal{H}_2 approximation errors obtained by the proposed model reduction method and the projection-based method [7] are given in Table I. It can be seen from this table that the proposed method has a clearly better performance in the sense of H_2 approximation error.

TABLE I: The \mathcal{H}_2 error comparison in Case 1

Approaches	Proposed Method	Monshizadeh [7]
$\ \Sigma - \hat{\Sigma}\ _{\mathcal{H}_2}$	0.312	0.393
$\ \Sigma\ _{\mathcal{H}_2}$		

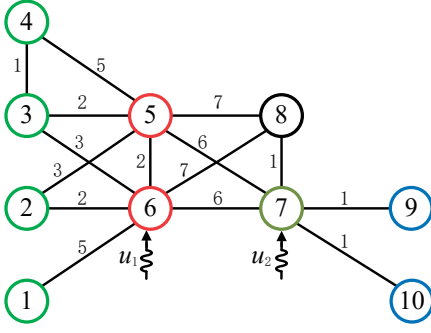


Fig. 2: An undirected network with 10 nodes, in which nodes 6 and 7 are leaders.

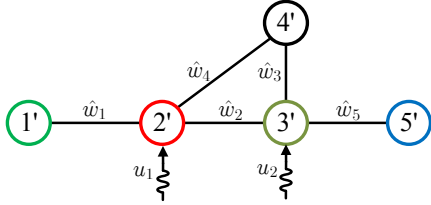


Fig. 3: The reduced undirected network with 5 nodes, in which nodes 2 and 3 are leaders.

TABLE II: The \mathcal{H}_2 error comparison in Case 2

Approaches	Proposed Method	Monshizadeh [7]
$\ \Sigma - \hat{\Sigma}\ _{\mathcal{H}_2}$	0.523	0.527
$\ \Sigma\ _{\mathcal{H}_2}$		

Case 2. Choosing the same output matrix as in [7]: $H = W^{\frac{1}{2}}B^T$, where $W \in \mathbb{R}^{15 \times 15}$ and $B \in \mathbb{R}^{10 \times 15}$ denote the edge weights and the incidence matrix of the original network in Fig. 2. Let $\beta = 1$. Using the almost equitable partition, Algorithm 1 then delivers a sub-optimal \hat{W} as $\hat{W} = \text{diag}(19.575, 12.0784, 0.89, 14.1376, 1.98)$. Thus, the reduced Laplacian matrix is obtained:

$$\hat{L} = \begin{bmatrix} 19.575 & -19.575 & 0 & 0 & 0 \\ -19.575 & 45.791 & -12.0784 & -14.1376 & 0 \\ 0 & -12.0784 & 14.9484 & -0.89 & -1.98 \\ 0 & -14.1376 & -0.89 & 15.0276 & 0 \\ 0 & 0 & -1.98 & 0 & 1.98 \end{bmatrix}.$$

The relative \mathcal{H}_2 approximation error is compared in Table II, showing that the reduced-order network system obtained by the proposed model reduction method gives a slightly better approximation of the original network system. Note that the approximation error obtained by [7] has been already very close to the optimum, and our method, which is not based on clustering-based projection, can also achieve similar approximation accuracy. With this output matrix and the almost equitable partition, it is rather difficult to acquire a better approximation.

VI. CONCLUSION

In this paper, the \mathcal{H}_2 model reduction problem for linear consensus networks has been formulated as a minimization problem, in which the edge weights in the reduced network are parameters to be chosen. Necessary and sufficient conditions have been proposed for constructing a set of optimal

edge weights. An iterative algorithm has been provided to search for the desired edge weights such that the \mathcal{H}_2 norm of the approximation error is small. Finally, compared with the projection-based method in [7], the feasibility of this method is illustrated by an example. The advantage of this proposed model reduction method is that not only the structure of the original network has been preserved but also the approximation error has been optimized. For future works, we will improve the effectiveness of the iterative algorithm such that the obtained solution is not restricted to a local optimum. Moreover, an extension to networked high-order linear subsystems is also of interest.

REFERENCES

- [1] Z. Li, Z. Duan, G. Chen, and L. Huang, "Consensus of multiagent systems and synchronization of complex networks: a unified viewpoint," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 213–224, 2010.
- [2] X. Cheng and J. M. A. Scherpen, "Clustering approach to model order reduction of power networks with distributed controllers," *Advances in Computational Mathematics*, vol. 44, no. 6, pp. 1917–1939, Dec 2018.
- [3] M. Mesbahi and M. Egerstedt, *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.
- [4] A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems*. Philadelphia, USA: SIAM, 2005.
- [5] X. Cheng, J. M. A. Scherpen, and B. Besselink, "Balanced truncation of networked linear passive systems," *Automatica*, vol. 104, pp. 17–25, June 2019.
- [6] A. J. van der Schaft, "On model reduction of physical network systems," in *Proceedings of 21st International Symposium on Mathematical Theory of Networks and Systems (MTNS)*, Groningen, The Netherlands, 2014, pp. 1419–1425.
- [7] N. Monshizadeh, H. L. Trentelman, and M. K. Camlibel, "Projection-based model reduction of multi-agent systems using graph partitions," *IEEE Transactions on Control of Network Systems*, vol. 1, pp. 145–154, Jun. 2014.
- [8] B. Besselink, H. Sandberg, and K. H. Johansson, "Clustering-based model reduction of networked passive systems," *IEEE Transactions on Automatic Control*, vol. 61, no. 10, pp. 2958–2973, Oct 2016.
- [9] P. Mlinarić, S. Grundel, and P. Benner, "Efficient model order reduction for multi-agent systems using QR decomposition-based clustering," in *Proceedings of 54th IEEE Conference on Decision and Control (CDC)*, Dec 2015, pp. 4794–4799.
- [10] T. Ishizaki, K. Kashima, A. Girard, J.-i. Imura, L. Chen, and K. Aihara, "Clustered model reduction of positive directed networks," *Automatica*, vol. 59, pp. 238–247, 2015.
- [11] X. Cheng, Y. Kawano, and J. M. A. Scherpen, "Reduction of second-order network systems with structure preservation," *IEEE Transactions on Automatic Control*, vol. 62, pp. 5026 – 5038, 2017.
- [12] —, "Model reduction of multi-agent systems using dissimilarity-based clustering," *IEEE Transactions on Automatic Control*, 2018.
- [13] C. Godsil and G. F. Royle, *Algebraic Graph Theory*. Springer Science & Business Media, 2013, vol. 207.
- [14] X. Cheng, Y. Kawano, and J. M. A. Scherpen, "Graph structure-preserving model reduction of linear network systems," in *Proceedings of 2016 European Control Conference (ECC)*, Aalborg, Denmark, 2016, pp. 1970–1975.
- [15] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.
- [16] G. Pipeleers, B. Demeulenaere, J. Swevers, and L. Vandenbergh, "Extended LMI characterizations for stability and performance of linear systems," *Systems & Control Letters*, vol. 58, no. 7, pp. 510–518, 2009.
- [17] M. C. de Oliveira and R. E. Skelton, "Stability tests for constrained linear systems," in *Perspectives in Robust Control*. Springer, 2001, pp. 241–257.
- [18] J. Lofberg, "Yalmip: A toolbox for modeling and optimization in MATLAB," in *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*. IEEE, 2004, pp. 284–289.